2. Explain the difference between a method declaration and a method body.

A method declaration is the first line that shows the method's name, return type, and parameters. The method body is the code inside the curly brackets that actually runs when the method is called.

---

3. What type of keyword is used to change the access level of a method?

You use access modifier keywords, like public, private, or protected.

---

4. What is another word used for describing the access level of a method?

Another word for access level is the visibility of the method.

---

5. Explain the scope of each variable in the code below.

var1: Only exists inside the main method.

var2: Only exists inside the for loop in main.

var3: Only exists inside the method1 method.

var4: Only exists inside the for loop in method1.

---

6. Write a method declaration for each description.

public static int getVowels(String word)

public static int extractDigit(int number)

public static String insertString(String text, int value)

7. a) How does the compiler distinguish one method from another?

The compiler looks at the method name and the number and type of parameters. This is called the method's signature.

b) Can two methods in the same class have the same name? Explain.

Yes, they can. This is called method overloading, and it works as long as the methods have different parameters.

---

8. a) What is the return statement used for?

A return statement sends a value back to the method that called it.

b) How many values can a return statement send back?

Only one value per return statement.

c) How is a method that returns a value different from one that doesn't?

A method that returns a value must have a return type and must use a return statement. A method that doesn't return anything uses the word void and doesn't return a value.

---

9. Find and explain the error in the code.

The error is that doSomething() is declared to return void, but the method tries to return(5);. A void method can't return a value. You either remove the return value or change the method to return an int.

---

11. True/False with explanations

a) True. Breaking a task into smaller methods is procedural abstraction.

b) False. A method call does not contain a method declaration.

c) False. A void method doesn't return a value.

d) False. Access modifiers don't declare return type; they set visibility.

e) False. static means the method belongs to the class, not an object.

f) False. Parameters are inside parentheses, not braces.

g) False. Local variables can only be used inside the method they're in.

h) True. A method can change the value of an argument if it's a reference type.

i) False. Overloading just means more than one method with the same name but different parameters.

j) True. The return statement sends a value back.

k) True. A precondition explains what must be true before the method runs.

l) True. A postcondition explains what the method does or guarantees after it runs.